

Claims

What is claimed is:

1. A method of providing automatic recovery from operating system faults, said method comprising the steps of:

- 5 detecting a system fault;
- analyzing the system fault;
- determining a cause of the system fault;
- determining a solution; and
- applying a solution.

10 2. The method according to Claim 1, further comprising the steps of:

- providing a resolution test; and
- returning to production.

3. The method according to Claim 1, wherein at least one of the recited steps does not require any work.

4. The method according to Claim 2, wherein at least one of the recited steps does not require any work.

5. The method according to Claim 1, wherein said detecting step comprises at least one of:

5 an operating system call to a halting routine; and

an exception or error associated with at least one of: an operating system, middleware, firmware and Licensed Internal Code.

6. The method according to Claim 1, wherein said detecting step comprises an abnormal termination of a driver or application.

10 7. The method according to Claim 1, wherein said detecting step comprises a hypervisor observation of unusual behavior from a guest operating system.

8. The method according to Claim 1, wherein said detecting step comprises an interception of a call to an operating system halting routine or exception handler.

9. The method according to Claim 1, wherein said detecting step comprises
15 automatically inspecting at least one aspect relating to the operating system.

10. The method according to Claim.9, wherein said detecting step comprises automatically inspecting at least one of: main memory; a kernel stack; process stacks; a state of all running threads; an amount of pageable memory used; an amount of pageable memory free for use; an amount of total pageable memory in the system; an amount of
5 total pageable memory available to the operating system kernel; an amount of non-pageable memory used; an amount of Non-pageable memory free for use; an amount of total non-pageable memory in the system; an amount of total non-pageable memory available to the operating system kernel; a number of system page table entries used; a number of system page table entries available for use; an amount of virtual memory
10 allocated to a system page table; a size of a system cache; a size of a page cache; a size of a file cache; an amount of space available in a system cache; an amount of space available in a page cache; an amount of space available in a file cache; a size of a system working set; a number of system buffers available; page sizes; a number of network connections established; utilization of one or more central processing units; a number of threads
15 allocated; a percentage of time spent in a kernel; a number of system interrupts per unit time; a number of page faults per unit time; a number of page faults in a system cache per unit time; a number of paged pool allocations per unit time; a number of non-paged pool allocations per unit time; a length of look-aside lists; a number of open file descriptors; an amount of free space on a disk or disks; a percentage of time spent at interrupt level; a

number of device drivers that are loaded; status of loaded device drivers; a number of outstanding I/O requests for device drivers; a state of devices attached to the system.

11. The method according to Claim 9, wherein said step of automatically inspecting comprises determining a degree of memory corruption.

5 12. The method according to Claim 11, wherein manual fault resolution is prompted if memory corruption is detected.

13. The method according to Claim 9, wherein said step of automatically inspecting is performed via software.

14. The method according to Claim 1, wherein said step of determining a cause
10 comprises identifying at least one faulty component.

15. The method according to Claim 14, wherein said analyzing step provides input into said step of determining a cause.

16. The method according to Claim 14, wherein external information provides input into said step of determining a cause.

17. The method according to Claim 1, wherein said step of applying a solution comprises effecting one or more changes or updates in at least one of: device driver software, operating system code, and firmware.

18. The method according to Claim 17, wherein said step of effecting one or
5 more changes or updates comprises deactivating faulty software.

19. The method according to Claim 2, wherein said step of providing a resolution test comprises monitoring a new component during a trial period.

20. The method according to Claim 19, wherein the trial period is over a finite period of time.

10 21. The method according to Claim 19, wherein the status of the new component is reported subsequent to the trial period.

22. The method according to Claim 21, wherein at least one of the following steps is repeated upon determination of a negative status of the new component:
detecting a system fault; analyzing the system fault; determining a cause of the system
15 fault; determining a solution; applying a solution; and providing a resolution test.

23. An apparatus for providing automatic recovery from operating system faults, said apparatus comprising:

an arrangement for detecting a system fault;

an arrangement for analyzing the system fault;

an arrangement for determining a cause of the system fault;

an arrangement for determining a solution; and

5 an arrangement for applying a solution.

24. The apparatus according to Claim 23, further comprising:

an arrangement for providing a resolution test; and

an arrangement for returning to production.

25. The apparatus according to Claim 23, wherein said detecting arrangement is
10 adapted to provide at least one of:

an operating system call to a halting routine; and

an exception or error associated with at least one of: an operating system,
middleware, firmware and Licensed Internal Code.

26. The apparatus according to Claim 23, wherein said detecting arrangement is adapted to provide an abnormal termination of a driver or application.

27. The apparatus according to Claim 23, wherein said detecting arrangement is adapted to provide a hypervisor observation of unusual behavior from a guest operating
5 system.

28. The apparatus according to Claim 23, wherein said detecting arrangement is adapted to provide an interception of a call to an operating system halting routine or exception handler.

29. The apparatus according to Claim 23, wherein said detecting arrangement is
10 adapted to automatically inspect at least one aspect relating to the operating system.

30. The apparatus according to Claim 29, wherein said detecting arrangement is adapted to automatically inspect at least one of: main memory; a kernel stack; process stacks; a state of all running threads; an amount of pageable memory used; an amount of pageable memory free for use; an amount of total pageable memory in the system; an
15 amount of total pageable memory available to the operating system kernel; an amount of non-pageable memory used; an amount of Non-pageable memory free for use; an amount of total non-pageable memory in the system; an amount of total non-pageable memory

available to the operating system kernel; a number of system page table entries used; a number of system page table entries available for use; an amount of virtual memory allocated to a system page table; a size of a system cache; a size of a page cache; a size of a file cache; an amount of space available in a system cache; an amount of space available in a page cache; an amount of space available in a file cache; a size of a system working set; a number of system buffers available; page sizes; a number of network connections established; utilization of one or more central processing units; a number of threads allocated; a percentage of time spent in a kernel; a number of system interrupts per unit time; a number of page faults per unit time; a number of page faults in a system cache per unit time; a number of paged pool allocations per unit time; a number of non-paged pool allocations per unit time; a length of look-aside lists; a number of open file descriptors; an amount of free space on a disk or disks; a percentage of time spent at interrupt level; a number of device drivers that are loaded; status of loaded device drivers; a number of outstanding I/O requests for device drivers; a state of devices attached to the system.

31. The apparatus according to Claim 29, wherein said detecting arrangement is adapted to determine a degree of memory corruption.

32. The apparatus according to Claim 31, wherein manual fault resolution is prompted if memory corruption is detected.

33. The apparatus according to Claim 29, wherein said detecting arrangement is adapted to perform automatic inspecting via software.

34. The apparatus according to Claim 23, wherein said arrangement for determining a cause is adapted to identify at least one faulty component.

5 35. The apparatus according to Claim 34, wherein said analyzing arrangement provides input into said arrangement for determining a cause.

36. The apparatus according to Claim 34, wherein external information provides input into said arrangement for determining a cause.

37. The apparatus according to Claim 23, wherein said arrangement for applying
10 a solution is adapted to effect one or more changes or updates in at least one of: device driver software, operating system code, and firmware.

38. The apparatus according to Claim 37, wherein said arrangement for effecting one or more changes or updates is adapted to deactivate faulty software.

39. The apparatus according to Claim 24, wherein said arrangement for providing
15 a resolution test comprises monitoring a new component during a trial period.

40. The apparatus according to Claim 39, wherein the trial period is over a finite period of time.

41. The apparatus according to Claim 39, wherein said arrangement for providing a resolution test is adapted to report the status of the new component subsequent to the
5 trial period.

42. The apparatus according to Claim 41, wherein at least one of the following is repeated upon determination of a negative status of the new component: detecting a system fault; analyzing the system fault; determining a cause of the system fault; determining a solution; applying a solution; and providing a resolution test.

10 43. A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for providing automatic recovery from operating system faults, said method comprising the steps of:

detecting a system fault;

analyzing the system fault;

15 determining a cause of the system fault;

determining a solution; and

applying a solution.